

To answer this question, we need to understand how a FIFO (First-In-First-Out) queue works and analyze each sequence to determine if it could be a valid output of enqueue and dequeue operations. Let's break it down:

1. FIFO Queue Behavior:

- Elements are added to the rear of the queue (enqueue).
- Elements are removed from the front of the queue (dequeue).
- The first element added is the first one to be removed.

2. Given Information:

- Enqueue operations insert integers 0 through 9 in order.
- Dequeue operations print the return values.

Now, let's analyze each sequence:

a. 0 1 2 3 4 5 6 7 8 9 This sequence is valid. It represents enqueueing all elements (0 to 9) and then dequeueing them all without any intermixing.

b. 4 6 8 7 5 3 2 9 0 1 This sequence is not valid. After 4 is dequeued, 5 must be at the front of the queue. It's impossible to dequeue 6 before 5.

c. 2 5 6 7 4 8 9 3 1 0 This sequence is not valid. After 2 is dequeued, 3 must be at the front of the queue. It's impossible to dequeue 5 before 3.

d. 4 3 2 1 0 5 6 7 8 9 This sequence is valid. It can be achieved by:

- Enqueue 0, 1, 2, 3, 4
- Dequeue 5 times (4, 3, 2, 1, 0)
- Enqueue 5, 6, 7, 8, 9
- Dequeue 5 times (5, 6, 7, 8, 9)

Therefore, the sequences that could not occur are: b. 4 6 8 7 5 3 2 9 0 1 c. 2 5 6 7 4 8 9 3 1 0

Let's analyze sequence **d** to verify whether it can be achieved with a FIFO queue:

Sequence d: 4 3 2 1 0 5 6 7 8 9

Steps to achieve the sequence:

1. Initial Enqueues:

- Enqueue 0
- Enqueue 1
- Enqueue 2
- Enqueue 3
- Enqueue 4

2. Dequeues:

- Dequeue: 4
- Dequeue: 3
- Dequeue: 2
- Dequeue: 1
- Dequeue: 0

At this point, the queue is empty, and the sequence is `[4, 3, 2, 1, 0]` .

3. Additional Enqueues:

- Enqueue 5
- Enqueue 6
- Enqueue 7
- Enqueue 8
- Enqueue 9

4. Additional Dequeues:

- Dequeue: 5
- Dequeue: 6
- Dequeue: 7
- Dequeue: 8
- Dequeue: 9

The final sequence is `[4, 3, 2, 1, 0, 5, 6, 7, 8, 9]` .